Programmable Logic Controllers and Direct Digital Controls in Buildings

Khalid. Y
Electronic and Electrical Engineering
University of Chester, UK
Email: y.khalid@chester.ac.uk

Abstract

The concept of programmable logic controller (PLC) originated over the last century that has revolutionised the industrial sector. In the last few decades PLC in the form of DDC has been commonly used in Building Energy Management Systems (BEMS). The contribution of this work is to analyse PLC/DDC role in the ongoing BEMS advancements in the building sector. Currently DDC are not understood by building design and simulation engineers who assess the controllability of the building in practice. This paper would enhance the understanding of integrating DDC in buildings and influence creation of better modelling and simulation tools for assessing their impact on energy performance in practice.

Keywords: Buildings, PLC, DDC, DCS, BEMS, BAS, Programmable Logic Controllers.

1. Introduction

Buildings are responsible for about 40% of electricity use in and contribute to approximately 30% of greenhouse gas emissions. Research over the past 20 years has led the design and construction of advanced low-energy buildings, which use novel energy generation and servicing technologies managed by BEMS to maintain comfortable internal conditions [1].

Modern buildings are complex multi-vector energy systems, with physical effects, multiple constraints and control/operational objectives. Control problems in the buildings industry are not trivial. However, the consequences of failure of plant through bad control are rarely catastrophic. The industry has been able to treat many problems through regular maintenance and commissioning schedules of BEMS [1].

This has sometimes led to surprisingly good results, but frequently fails to satisfy all the essential occupant's comfort, energy use, operating cost and capital cost requirements and hence defeating the whole purpose of BEMS & BAS [1].

2. Building Energy Management Systems (BEMS)

BEMS is a high technology control system installed on buildings which performs the overall control and monitoring functions for some or all of the building's plant and systems (mechanical and electrical equipment such as air handling, cooling plant systems, lighting and power systems etc.).

In modern buildings, BEMS is implemented as a networked direct digital control system (shown below) consisting of both software and hardware (PLCs), display terminals with user interface for system scheduling and control.

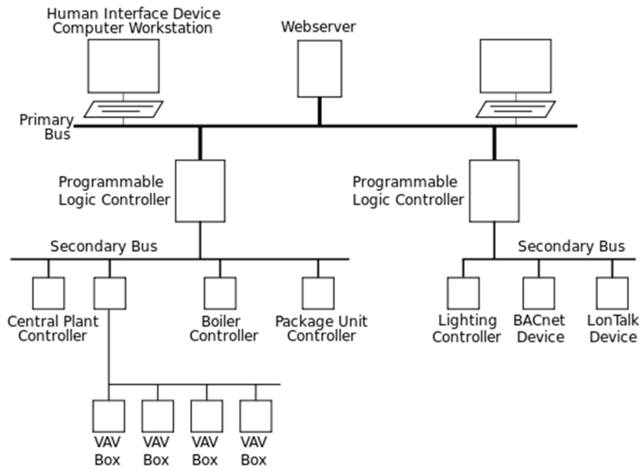


Figure 1 Schematic of BEMS [2]

The whole distributed control system has a hierarchical topology where communication between energy system and controls is carried out by PLC's through standard building automation communication protocols such as C-Bus and internet enabled protocols such as BACNet and LonWorks. Current control functions/strategies (such as scheduling, resets, lockouts and sequencing/staging) performed by BEMS are summarised in [1]. In most cases several of these PLCs are used and operated in loops simultaneously and this has presented control systems designers the challenge to design, commission and operate advanced BEMS to successfully manage all building services.

3. Basic Technology

Electrical Engineering is a discipline concerned with generation and transmission of electrical power where as electronic engineering is focused on manipulation and control of electrical power supplied to electrical machines for the end user. The purpose of control systems is to control the operational state (function) of electrical devices by switching/controlling the electrical supply. Before the advent of PLCs, control systems were implemented using vacuum tubes however because of their inefficiency these were superseded with semi-conductor devices such as transistors and diodes as used today in digital electronic circuits [3].

In digital electronics, physical signals such as temperature, pressures etc. are analogue signals converted into electrical signals and digitised (or quantised) into logic levels (shown below) before being stored as binary data in computer systems.

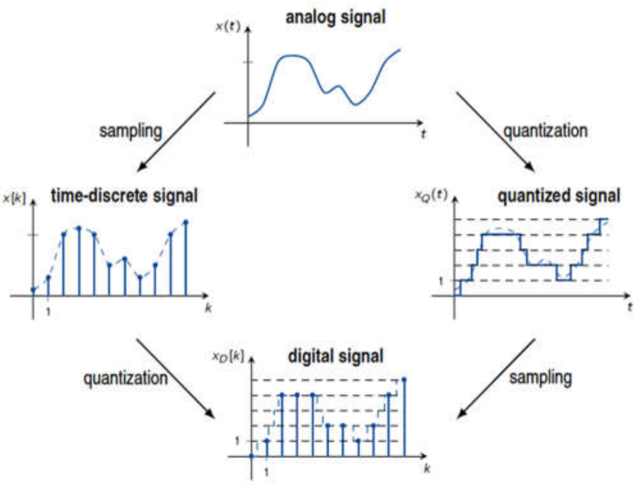


Figure 2 Analogue to digital conversion [4]

Manipulation of binary data is governed by logic functions that are created using Boolean functions used for representing circuits in the form of mathematical equations [5].

The Boolean functions are implemented as logical operations by logic gates (figure 3) having binary inputs and outputs. Logic gates are implemented using diodes or transistors (figure 4) acting as electronic switches to create binary signals. Generally, these gates are manufactured from field effect transistors (FETs) and metal-oxide-semiconductor field effect transistors (MOSFETs).

These logic gates are building blocks of all digital devices and modern microprocessors may contain more than 100 million logic gates [6].

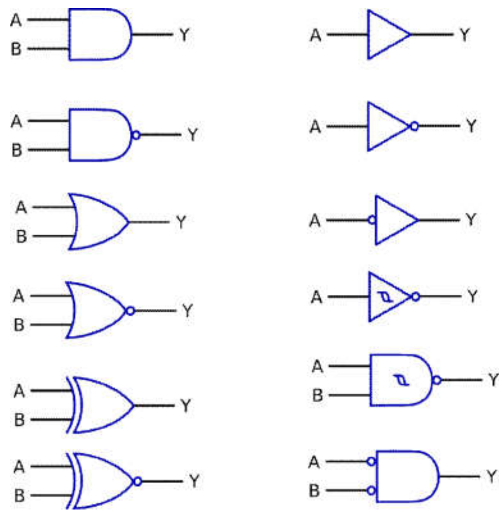


Figure 3 Logic gates

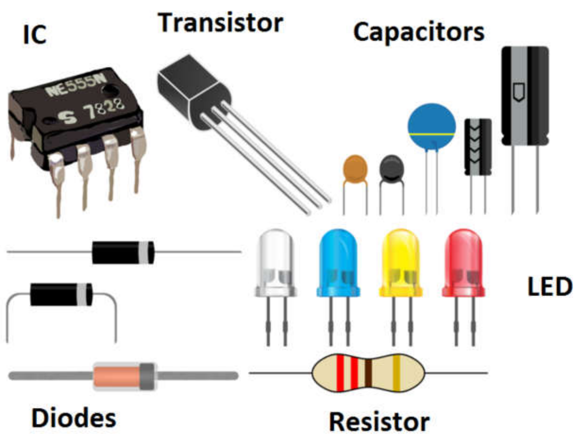
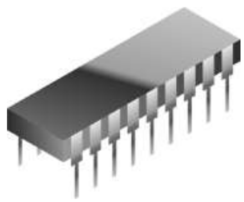


Figure 4 Transistors and Diodes

Multiple logic gates are packaged together to create integrated circuits (IC) which perform basic functions such as counting, memory and arithmetic. These integrated circuits are soldered on to printed circuit boards (PCB) [7] and utilised in all devices today as electronic chips.



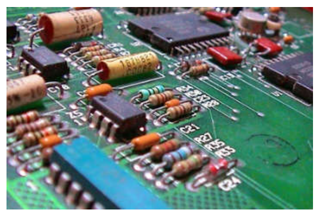


Figure 5 IC (left) Figure 6 PCB (right)

We can find electronic chips that perform independent functions such as: memory module, and processor chips used in normal home PCs (figure 7).

Where all functions such as counting, memory and arithmetic processing are all embedded on a single electronic chip then these are generally known as micro-controllers (figures 8 & 9).

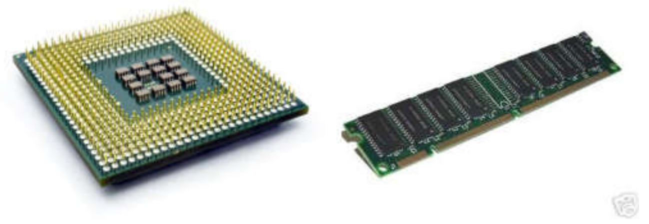


Figure 7 Microprocessor (left) and Memory (right)

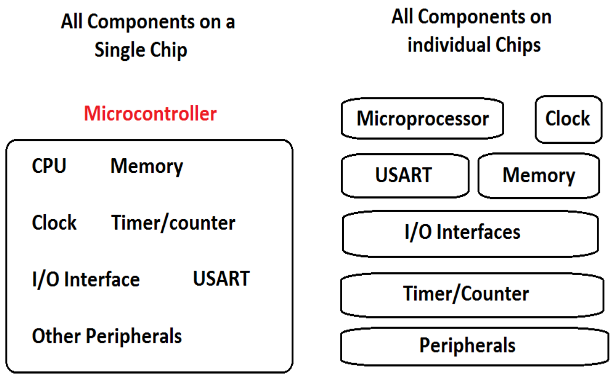


Figure 8 Classification of chips [8]

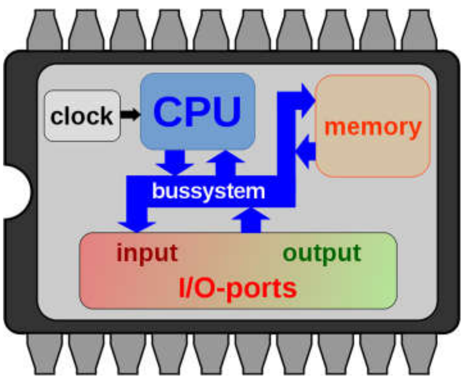


Figure 9 Micro-Controller Internal structure

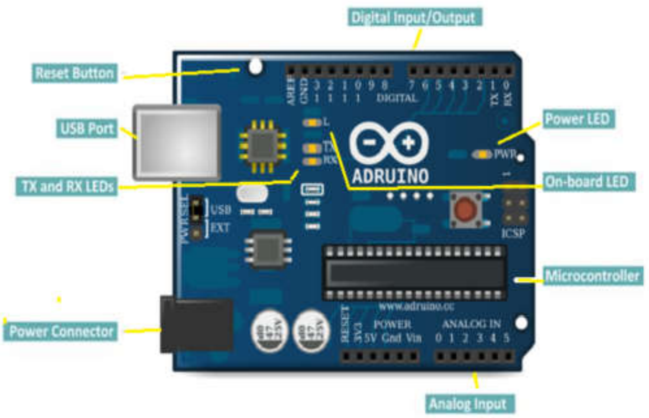


Figure 10 Arduino microcontroller

The terms Microcontroller and Microprocessor are sometimes confused and used interchangeably. In a microcontroller, the processor core, memory, and programmable Input-Output (IO) peripherals are integrated into a single System on Chip as shown in figures (9).

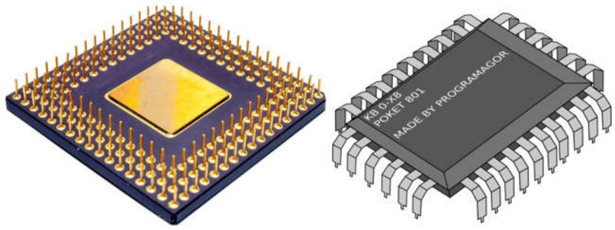


Figure 11 Microprocessor (left) and Microcontroller (right)

Microprocessors, on the other hand, may have integrated registers, but rely on external RAMs and peripherals i.e. also known as a CPU [9].

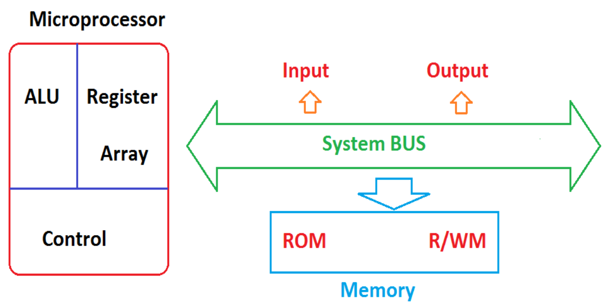


Figure 12 Microprocessor structure

Unlike microcontrollers, PLCs have modular structure with three core area: Power supply, microprocessor based CPU & I/O. PLC shares typical features of home computer having CPU, memory, software and communication interface (I/O) [10].

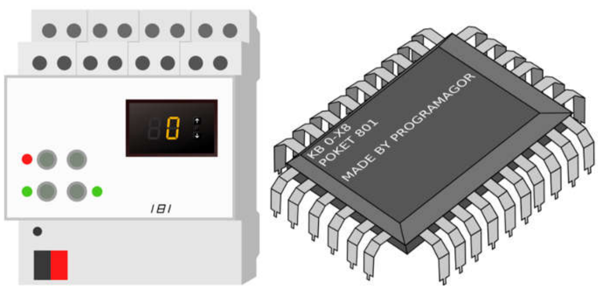


Figure 13 PLC (left) and microcontroller (right)

PLC generally work on high voltage AC power for activating/supplying mains power and using CPU and communication modules for implementation of logic, sequencing, timing and counting and arithmetic. Whereas the microcontrollers work on low voltage DC power, and use transistors to control the power according to the design and operation of the machine or specific application.

4. Programmable Logic Controller (PLC)

The PLC is a dedicated controller designed for large industrial application or projects. The overall system is modular and the user need to add modules to increase the inputs/outputs (I/O), Power supply, microprocessor based CPU:

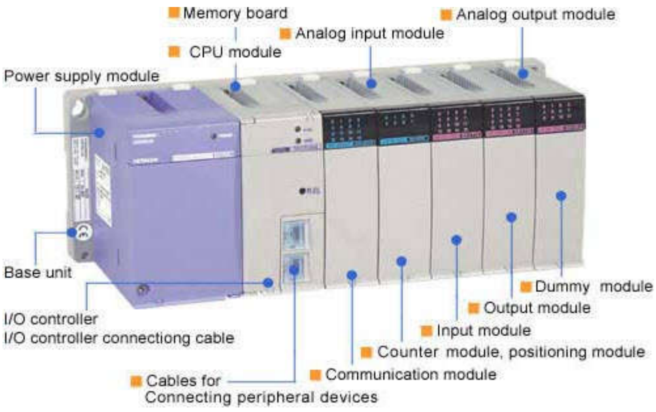


Figure 14 Modular Structure of typical PLC [11]

PLCs are devices on which control instructions and logic could be stored as a program and executed by operating the PLC with supplied electricity. The program could be for example; logical arguments such as if button is pressed then open door etc.

The PID or ON/OFF control algorithms, system sequencing instructions, system scheduling, counters, arithmetic processing are coded/programmed on the PLC, utilising the inputs (sensors) and outputs (actuation signals to motors, drives, solenoids etc.) to implement the control strategy for the particular system/device as shown below in figure 15.

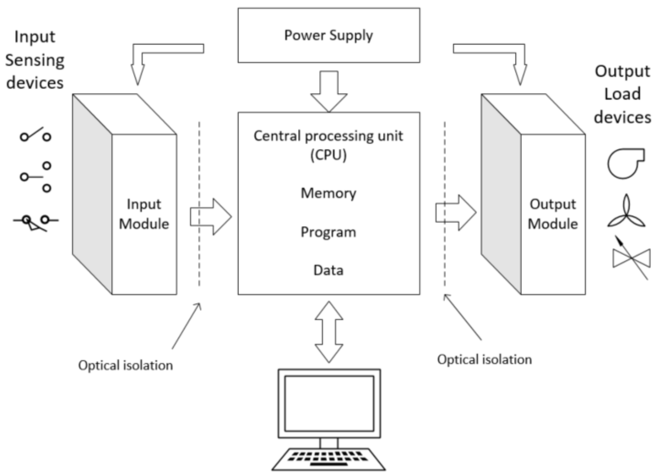


Figure 15 Basic Internal Structure of PLC

The structure of a PLC can be broken down into four basic components: Input interface, CPU, memory unit and output interface (figure 15). A programmable logic controller will continually "loop" through its internal "user defined" programme, waiting for inputs and giving outputs at the programmed specific times. The inputs to the PLC control panel come from operator panel inputs (buttons) or field equipment inputs i.e. sensors. The outputs (i.e. actuators) can be displayed on an operator panel (indicators) or activate equipment in the field as shown below.

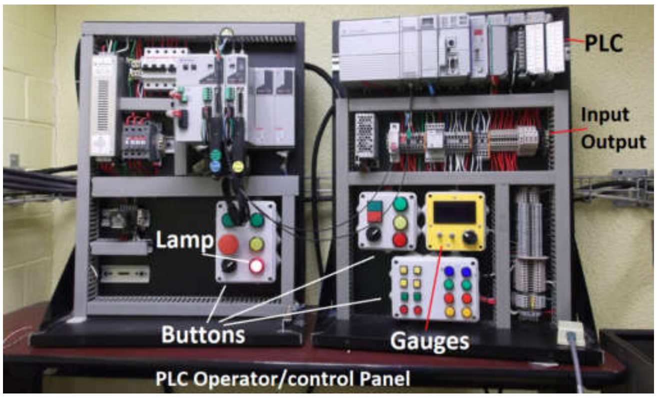


Figure 16 Internal structure of PLC control system

The control system can be graphically monitored and "data logging" is possible in more advanced systems using the SCADA (supervisory control and Data Acquisition) software platform to monitor, control and acquire data from field devices. SCADA is normally a software package designed to display information, log data and show alarms in a graphical interface showing the system.

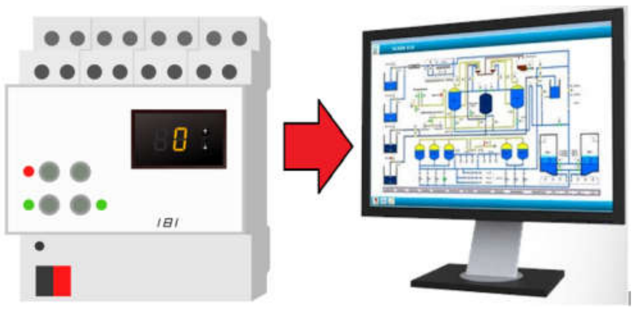


Figure 17 PLC and SCADA

PLC’s are programmed with user instructions using various programming languages as specified by the PLC IEC 61131 standard (part 3) [5].

It defines two graphical, textual and one mix programming language [5]:

1. Ladder diagram (LD), graphical
2. Function block diagram (FBD), graphical
3. Structured text (ST), textual
4. Instruction list (IL), textual
5. Sequential Function Chart (SFC), mixture

5. PLC programming

Ladder diagram (LD) or Ladder Logic: LD represents a program by a graphical diagram based on the circuit diagrams of digital logic gates. LD is used to develop software for PLCs used in process control applications.

Functional block diagram (FBD): It is a graphical dataflow programming language for PLC design that can describe the function between input and output variables.

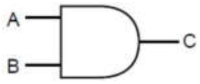
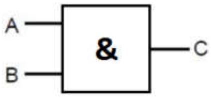
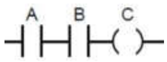
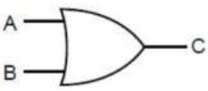
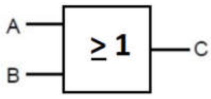
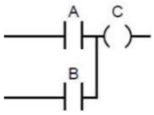
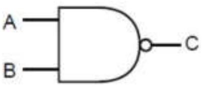
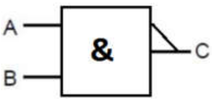
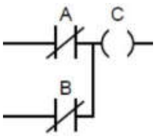
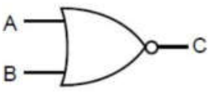
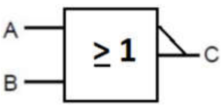
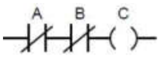
Logic Diagram	Functional block FBD	Ladder Diagram
 AND gate		
 OR gate		
 NAND gate		
 NOR Gate		

Table 1 LD representation of logic gates

Structured text (ST), textual: it is a high level language and is based on Pascal programming language:

Example code:

```
IF vessel.temp > 50 THEN
        Valve.on := 1;
        Valve.off := 0;

ELSE if vessel.temp > 20 THEN
        Valve.on:=0;
        Valve.off:=1;
END_IF
```

Instruction list (IL): It is an assembly like low level language where instructions, function calls and subroutines with optional parameters are used to execute program control.

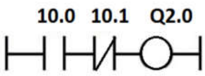
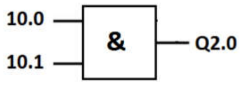
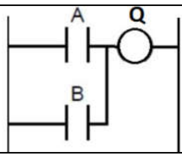
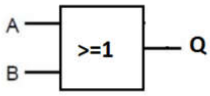
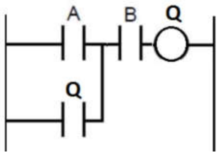
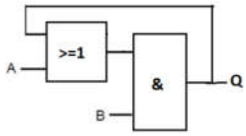
LD	FBD	IL
		LD I0.0 AN I0.1 = Q2.0
		LD A O B = Q
		LD A O Q LD B ALD = Q

Figure 18 IL compared to LD and FBD

Sequential Function Chart (SFC): SFC has features to execute programs for sequential and parallel control processing and the programs can be divided into steps. Steps consist of instructions that are carried out to satisfy a condition to allow transition to the next step, thus complex operations are divided into smaller segments.

The basic elements of the SFC are steps with action blocks and transitions:

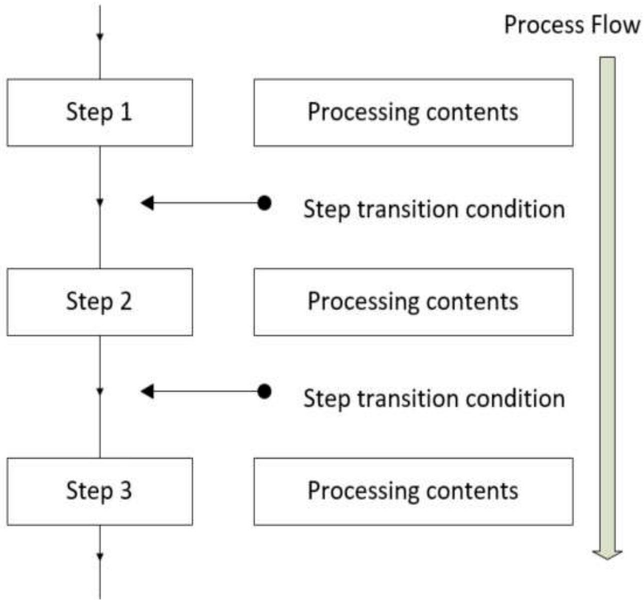


Figure 19 schematic of SFC elements

Each element can be programmed in any of the IEC language such as LD or ST or IL or FBD or mix of match of any of the languages.

6. Direct Digital Controls (DDC)

PLC (Programmable Logic Controller) is a small hardware device. It can be a combination of logic gates, a micro-controller, a simple CPU and analogue/digital interfaces.

DDC (Direct Digital Control) is a term used in building automation for the concept of reading sensor data, processing the data in the digital domain, and controlling the actuators intelligently.

Hence DDC are microcontroller based where all components are situated on one chip and digitally controlled using Analogue/digital data converters and multiplexers as shown in following figures.

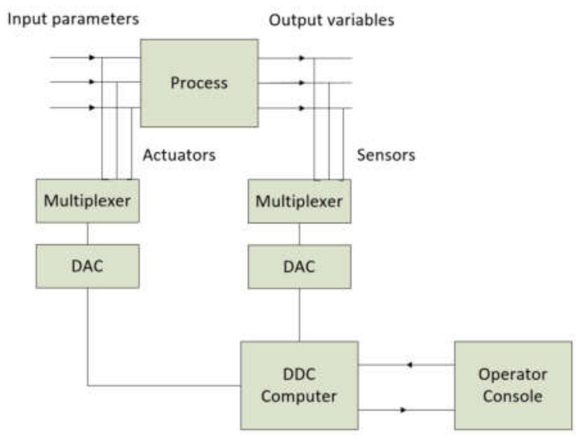


Figure 20 DDC controller functional diagram

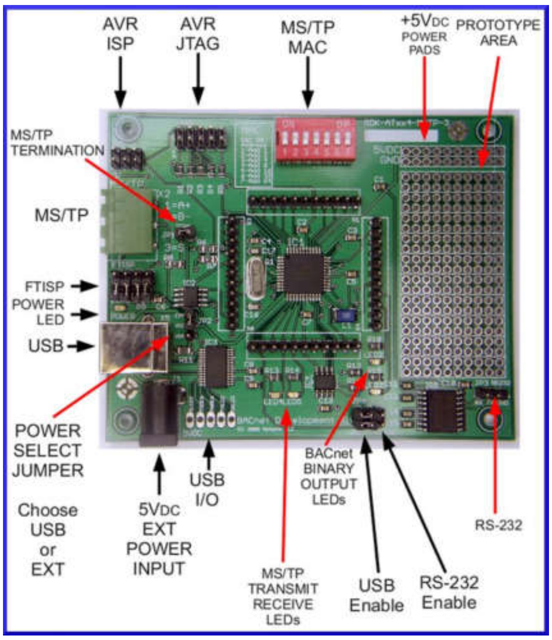


Figure 21 Structure BACnet Controller [12]

DDC technology is used to implement complex controls in buildings for managing all servicing systems (HVAC, security and lighting):

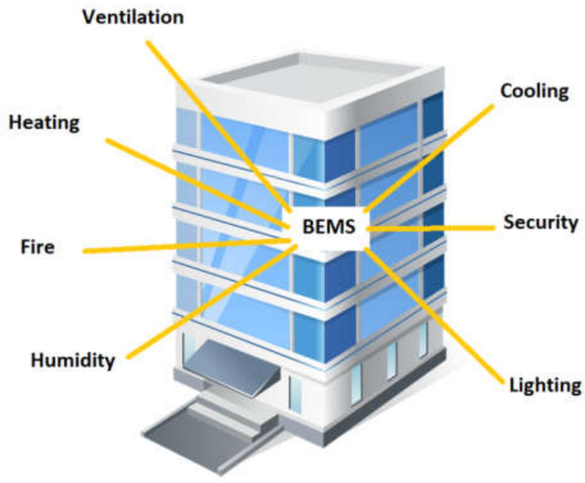


Figure 22 Large building schematic with BEMS

For today's modern buildings, typical BEMS/BAC hardware DDC is manufactured by companies such as TREND, SEIMENS etc. In reality a large building will have hundreds of different controllers operating – BEMS is a means of monitoring and coordinating their operation [1].

Figure 23 BACnet Building Controller [13]

The main advantage of PLC IEC standard is that it enables the programmer to use multiple programming languages in the same PLC thus different tasks are efficiently implemented using the most appropriate language. Alternatively the DDC controllers are designed to specific building applications and have bespoke OEM programming tools (figure 24).

The building automation DDC manufacturers have created their own proprietary software to make modifications to the program using their specific control language. Hence for recommissioning, troubleshooting and rescheduling of BEMS controllers requires trained engineers and the building owner to pay enormous maintenance fees to keep the BEMS system running efficiently [14].

Hence DDC technology poses many challenges for commissioning, rescheduling and coding of the controllers.

Figure 24 BAS state of the art controllers with algorithm programming facility [15]

Further complexity is when control system strategies are implemented. In a large building multiple different controllers controlling HVAC and lighting are installed by several DDC engineers from various manufacturers. Ideally DDC controllers in building need to be programmed to take account of underlying building physics, system dynamics and cross coupled nonlinear effects between the servicing systems / other controllers. Lack of knowledge of the above and collaboration and poor programming would lead to poor commissioning which is major reason for poor building energy performance.

Additionally, even if the above phenomena is somewhat taken into account, the basic control loop algorithms such as on/off and PID are implemented rather than intelligent algorithms. These basic algorithms cannot effectively deal with dynamic physical effects mentioned above leading to sluggish an oscillatory controller loops. Normally DDC devices have functionality for controller gain scheduling and auto-tuning techniques to mitigate the effects (figure 24).

Further advances in IT networking and communication technology are making it easier to set up building DDC control systems for creating DCS (distributed control systems) with supervisory network controls as well as device level DDC controllers. For example interoperability, standard communication protocols, and ad-hoc networking concepts are making the idea of plug-and-play a reality for DDC control devices on DCS network.

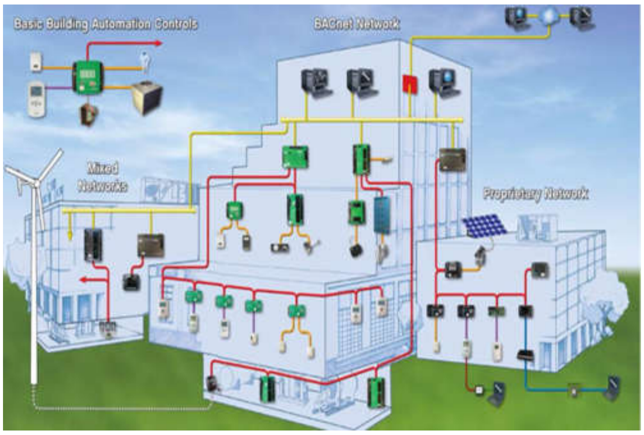


Figure 25 Automation in a Building [16]

Gaining access to information on a DDC controls network is also becoming much easier with Internet, wireless connectivity and sensing. The technologies have enabled data-mining and analysis of energy information in the building. This is a supervisory layer of the DCS system network for energy managers who have to interpret the building energy system behaviour with all above complexities of a DDC network.

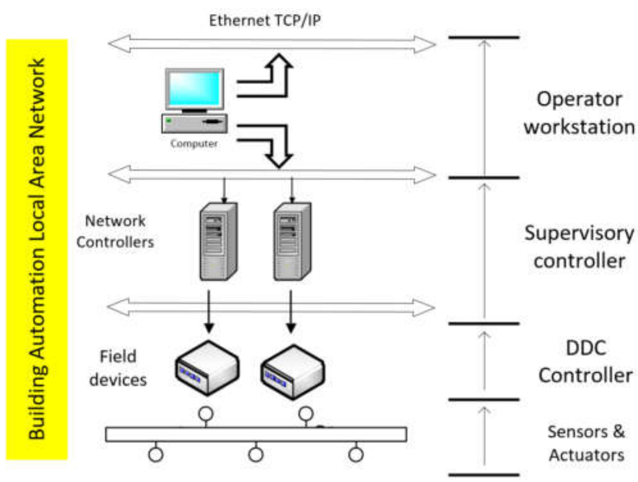


Figure 26 Distributed Control System (DCS)

Local level DDC controllers which control field devices are classed into three types:
1) Dedicated controllers
2) Programmable controllers
3) Fully programmable controllers

1) Dedicated controllers: Designed by OEM for specific HVAC application, e.g.: Fan Coil Units (FCU) and Variable Air Volume (VAV).

Figure 27 VAV controller [17]

2) Programmable controllers: Designed for specific HVAC equipment such as heat pumps, rooftop units with capability to program the controls.

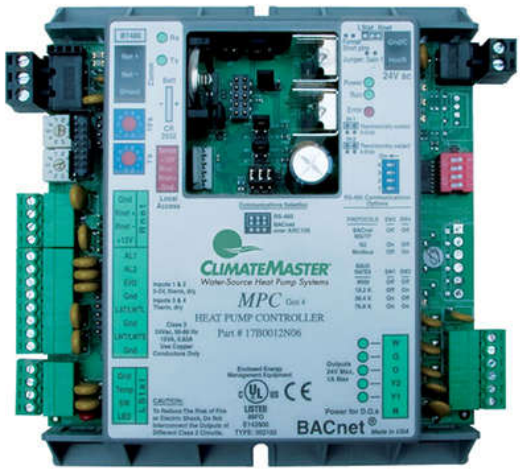


Figure 28 Heat pump controller [18]

3) Fully programmable controllers: Designed for flexibility and can be programmed to control HVAC plant equipment such as boilers, chillers, pumps for sequencing, scheduling, and control systems integration.

Figure 29 Flexible Plant controller [19]

Sometimes DDC controllers are used for supervisory/ building/systems level control in a DCS network rather than specific control of HVAC equipment. In this configuration the DDC controllers manage/share information between lower level group of controllers and global controllers. This configuration is referred to as P2P Networking (Peer-to-Peer).

On a DCS network, LAN communication is done by connecting controllers together for which there are different physical arrangements. The common ones are: Bus (or Daisy Chain) and Star networks.

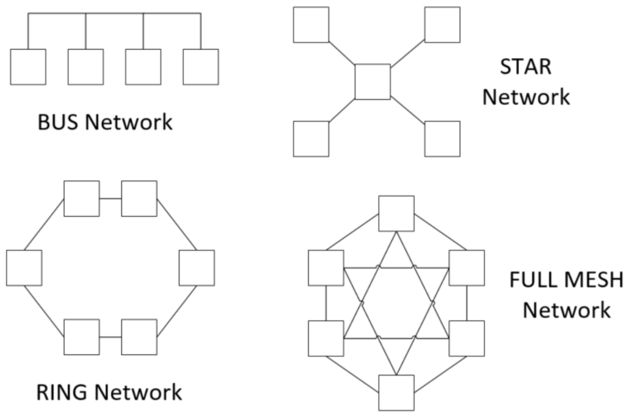


Figure 30 DDC controller configuration

7. DDC programming

There are three different approaches to writing the software programs in DDC controllers.

1. Text based programming: it is based on Basic/C programming language program. A sample of code is shown below:

```
1. IF [OAT;CV]<=20.0 THEN 2 ELSE 5
2. [TANK PUMP;CV]=1
3. SWAIT 30
4. [TANK START;CV]=1
5. IF [OAT;CV]>=70.0 THEN 6 ELSE 1
6. [TANK START;CV]=0
7. [TANK PUMP;CV]=0
```

Where:
OAT = outside air temperature
CV = control variable

2. Object oriented programming: Consists of library of predefined objects/logic-blocks/symbols representing various HVAC equipment and the control program is created by linking them together based on knowledge of electronic circuits.

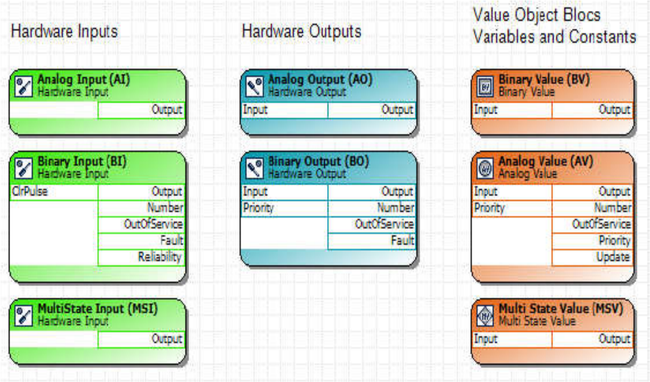


Figure 31 Block diagram language for DDC [20]

3. Application programming: the programmer is asked questions and based on the answers the controller program is generated. For example:
What type of Heat pump is this?
Press 1. Electric OR 2. Gas fired

8. Open Programming language - Sedona

Currently original equipment manufacturers prefer to create control algorithms and embed them on the on board electronics requiring their own programming languages for competitiveness. To avoid this, DDC controllers with standard control language was created using Sedona Framework based on component orientated programming similar to Java and C++. Sedona Framework with standard programming language and control systems design and development software has allowed designers to create networks of DDC controllers enabling smart control systems integration in buildings [21].

Using Niagara Workbench or a Sedona tool, such as Contemporary Controls' Sedona Application Editor (SAE), components are assembled onto a wire sheet, configured and then interconnected with other components to create control systems applications. This language allows graphical representation of control strategies using block diagrams where program changes are immediately executed [23].

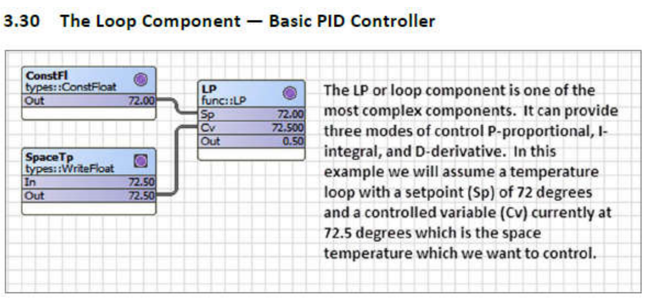


Figure 32 Sedona block diagram [22]

In Sedona programming language the programmer can use a library of control elements to create any control algorithm. The main advantage is that the Sedona language is not dependent on a proprietary software tool and thus Sedona enabled controller can be programmed using variety of tools and the controller code can be reused on a different Sedona enable controller [21].

Similar to a normal DDC controller, a Sedona enabled controller of standard hardware components such as CPU, memory and I/O interface. Additionally it consists of Sedona Virtual Machine (SVM) which resides on the controller and executes the Sedona application/control program created by the programmer [24].

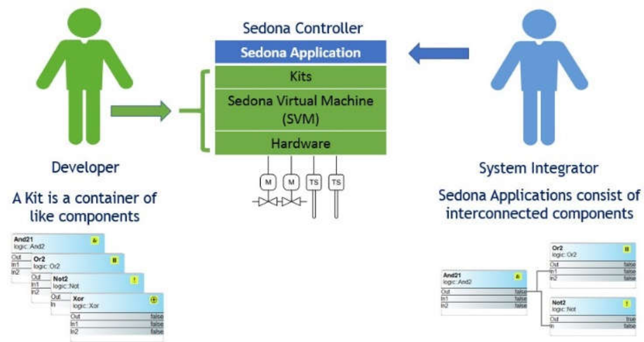


Figure 33 Sedona integration [24]

Currently Sedona application editor (SAE) is available free through the Contemporary Control website: https://ccontrols.com/tech/sedona.htm. A number of companies are selling Sedona compliant DDC products such as Tridium.

9. Conclusion

PLC are fundamental devices used today in almost every industry for controlling machinery. PLC is more reliable, robust and fast compared to DDC. However DDC controls are considered more appropriate in building automation and control systems due to their easier configuration of control loops and computer based analytics.

Currently the buildings sector lacks understanding of how BEMS and BAS work and the PLC and DDC are considered as black art which only the BMS engineer have the knowledge. It is important that workings of these devices are understood and awareness is raised to program them with understanding of the underlying physics, nonlinearities and system dynamics in order to improve building energy performance in practice.

The quantification of BEMS, DDC, BAS, DCS, PLC need robust modelling and simulation tools which can accurately compute their energy performance. Currently building simulation tools model the building very accurately, however they lack the simulation capabilities to mimic controller design and virtual commissioning of DDC and thus there is gap between how controls performance is assessed in building simulation and how they are design, installed and commissioned. Hence there is a building energy performance gap which is widely understood as due to inaccurate virtual performance assessment of control system in buildings due to lack of knowledge of these devices in building simulation experts and professionals.

Hence this paper's contribution is to bring together difficult concepts of PLC and DDC from design to practice, and programming to operation. So that building automation, control and simulation engineers in practice and research engineers across industry and academia can benefit from their holistic understanding in relation to controllability of buildings. With the aim of influencing changes in building control systems design process, advancement in modelling and simulation tools as well as commissioning in practice.

This work presents an analysis of PLCs with a view of understanding their role in the ongoing Building automation trends. In future, several aspects of PLC's related to automation processes will be studied and findings will be published as contributions to the ongoing research in this area.

10. References

[1] Khalid, Y. (2018). Controllability of buildings: computing and managing energy in practice. Journal of Computing and Management Studies (JCMS), 2(2).

[2] Attribution: Matthew Beckler at English Wikipedia

[3] Mark John Walker PhD thesis (2012), The programmable logic controller: its prehistory, emergence and application, The open University

[4] Attribution: Sascha Spors, Digital Signal Processing - Lecture notes featuring computational examples, 2016-17

[5] W Bolton, Programmable Logic Controllers, 5th Edition, 2009

[6] Flynn, M.J.; Hung, P. "Microprocessor design issues: thoughts on the road ahead". IEEE Micro, Vol. 25, Issue 3, May-June 2005 pp. 16 - 31.

[7] R.S.Khandpur, 2006, Printed Circuit Boards: Design, Fabrication, and Assembly (McGraw-Hill Electronic Engineering)

[8] Kumar, Saravanan, Jeevananthan, 2010; Microprocessors and Microcontrollers OUP India; Pap/Cdr edition (9 Dec. 2010)

[9] Nagoorkani (2012) Microprocessors & Microcontrollers, Tata McGraw-Hill Education

[10] PLC Handbook, Practical Guide to Programmable Logic Controllers, AutomationDirect.com

[11] http://seldomindia.com/plc-brief-description/

[12] BACNET Development Kit http://bacnetdevelopmentkit.com/ATMegaTechnical.htm

[13] TREND Data sheet IQ4 NC Controllers

[14] http://www.automatedbuildings.com/news/aug11/articles/ksin/110720111001ksin.html

[15] Aistier: Building Automation & Energy Management Solutions http://aistier.com/images/sub2/09_.pdf

[16] KMC controls Brochure: Controlling your green Building

[17] http://www.advancedair.co.uk/products/vav-units/model-series-3000-single-duct-vav

[18] http://www.climatemaster.com/commercial/?cm_product=mpc-and-lon-heat-pump-controls

[19] https://buildingcontrols.honeywell.com/HVAC-Controllers/EagleAX-Plant-Controller

[20] http://www.distech-controls.com/en/ca/training/

[21] https://www.ccontrols.com/pdf/wp/WPSEDONAAA0.pdf

[22] https://ccontrols.com/pdf/RM-SEDONA00.pdf

[23] https://ccontrols.com/tech/sedona.htm

[24] https://sedona-alliance.org/aboutsedona.htm